УДК 681

В.А. КОЛОСОВ, аспирант НТУ «ХПИ» (г. Харьков)

АНАЛИЗ КРИПТОСИСТЕМ С ОТКРЫТЫМ КЛЮЧЕМ

У статті приведений аналіз криптосистем з відкритим ключем, приведені алгоритми найбільш поширених систем, таких як RSA, Ель Гамаля, Діффі–Хельмана і криптосистема на основі еліптичних рівнянь. А також розглянуті сфери їх застосування.

In article the analysis of cryptosystems with an open key is resulted. Algorithms of the most widespread systems, such as RSA, by El Gamalja, Diffi-Hellmana and a cryptosystem on the basis of elliptical equations are resulted. And also areas of their application are considered.

Постановка проблемы. Криптосистема с открытым ключом была предложена Диффи и Хеллманом в 1976 г.[2,3] Эта работа является радикальным шагом в криптографии. С одной стороны в криптографических алгоритмах с открытым ключом используются математические функции, отличные от подстановок и перестановок. Но более важно то, что методы криптографии с открытым ключом являются асимметричными, т. е. предполагают использование двух различных ключей в отличие от методов симметричного традиционного шифрования с помощью одного ключа. *Первый ключ* является *открытым* и может быть опубликован для использования всеми пользователями системы для шифрования данных. Для расшифровки данных получатель зашифрованной информации использует *второй ключ*, который является *секретным*.

Диффи и Хеллман сформулировали требования, выполнение которых обеспечивает безопасность асимметричной криптосистемы.

1. Вычисление пары ключей (открытого и закрытого) получателем на основе начального условия должно быть простым.

2. Отправитель, зная открытый ключ и сообщение, может легко вычислить криптограмму.

3. Получатель, используя секретный ключ и

криптограмму, может легко восстановить исходное сообщение.

4. Противник, зная открытый ключ и криптограмму, при попытке вычислить секретный ключ

или исходное сообщение сталкивается с трудноразрешимой вычислительной проблемой.

В криптографических системах с открытым ключом используются так называемые *необратимые или односторонние функции*, которые обладают следующим свойством: при заданном значении x относительно просто вычислить значение $f(x)$, однако, если $y = f(x)$, то нет простого пути для вычисления значения x.

Под *необратимостью* понимается не теоретическая необратимость, а практическая невозможность вычислить обратное значение, используя со-

временные вычислительные средства, за обозримый интервал времени. Множество классов необратимых функций и порождает все разнообразие систем с открытым ключом. Все предлагаемые сегодня криптосистемы с открытым ключом основаны на одном из следующих типов необратимых преобразований: 1) разложение больших чисел на простые множители; 2) вычисление логарифма в конечном поле; 3) вычисление корней алгебраических уравнений.

Цель статьи Анализ распространенных криптографических систем с открытым ключом. Основные принципы асимметричных алгоритмов работы и области применения конечных реализаций в шифровании.

Основная часть. Следует отметить несколько распространенных заблуждений, касающихся шифрования с открытым ключом[2-4, 7].

- Шифрование с открытым ключом защищено от криптоанализа более традиционного шифрования. На самом деле защита любой схемы шифрования зависит от длины ключа (1) и от объема вычислительной работы, необходимой для взлома шифра (2).

- Шифрование с открытым ключом является универсальным, делающим традиционное шифрование устаревшим. Напротив, ввиду очень высоких требований, предъявляемых со стороны схем шифрования с открытым ключом к вычислительным ресурсам, отказ от схем традиционного шифрования является маловероятным.

- Распределение ключей при шифровании с открытым ключом является тривиальной задачей по сравнению с системами, построенными на традиционном шифровании. На самом деле в случае шифрования с открытым ключом необходим специальный протокол, нередко предполагающий существование некоторого центрального агента, а применяемые при этом процедуры не являются ни более простыми, ни более эффективными, чем те, которые требуются для традиционного шифрования.

Скорость шифрования в асимметричных криптосистемах намного ниже скорости шифрования в одноключевых, поэтому асимметричные криптосистемы используют только в двух случаях: 1) для шифрования секретных ключей, распределяемых среди пользователей вычислительной сети; 2) для формирования цифровой подписи.

Основное преимущество асимметричных алгоритмов перед симметричными состоит в том, что секретный ключ, позволяющий расшифровывать всю получаемую информацию, известен только приемнику. Кроме того, первоначальное распределение ключей в системе не требует передачи секретного ключа, который может быть перехвачен нарушителем.

Концептуальная схема шифрования с открытым ключом. Суть асимметричных криптосистем состоит в том, что каждым адресатом генерируются два ключа, связанные между собой по определенному правилу. Один ключ объявляется *открытым*, а другой – *закрытым*. Открытый ключ публикуется и доступен любому, кто желает послать сообщение адресату. Секрет-

ный ключ сохраняется в тайне. Исходный текст шифруется открытым ключом адресата и передается ему. Зашифрованный текст не может быть расшифрован тем же открытым ключом. Дешифрование сообщения возможно только с использованием закрытого ключа, который известен только самому адресату (рис. 1).

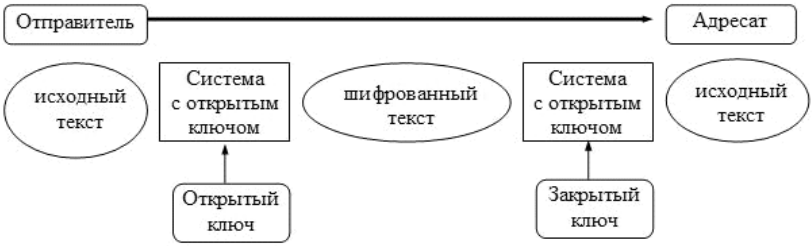

Рис. 1

Ниже рассматриваются наиболее распространенные системы с открытым ключом.

Криптосистемы с открытым ключом

Криптосистема *RSA*

Криптосистема *RSA*, предложенная в 1977 году Ривестом, Шамиром и Адлеманом, предназначена для шифрования и цифровой подписи. В настоящее время *RSA* является наиболее распространенной криптосистемой – стандартом де-факто для многих кирптографических приложений.

Авторы воспользовались тем фактом, что нахождение больших простых чисел в вычислительном отношении осуществляется легко, но разложение на множители произведения двух таких чисел практически невыполнимо. Доказано, что раскрытие шифра *RSA* эквивалентно такому разложению. Поэтому для любой длины ключа можно дать нижнюю оценку числа операций для раскрытия шифра, а с учетом производительности современных компьютеров оценить и необходимое на это время.

Возможность гарантированно оценить защищенность алгоритма *RSA* стала одной из причин популярности этой криптосистемы на фоне десятков других схем. Поэтому алгоритм *RSA* используется в банковских компьютерных сетях, особенно для работы с удаленными клиентами (обслуживание кредитных карточек).

В настоящее время криптосистема *RSA* широко применяется в составе различных стандартов и протоколов *Internet*, включая *PEM*, *S/TIME*, *PEM-MIME*, *S-HTTP* и *SSL*.

Недостатки существующего не одно десятилетие метода полной матрицы хорошо известны. Протокол Диффи-Хеллмана предполагает двусторонний обмен открытыми ключами и наличие сертификатов у отправителя и по-

лучателя сообщений. В случае односторонней аутентификации (сертификат имеется только у одной из сторон) предпочтение отдается последнему методу. В этой ситуации выбор *RSA* вполне оправдан – метод цифрового конверта на базе криптоалгоритма *RSA* описан в стандарте *PKCS* и применяется в протоколе *SSL* и других стандартах *Internet* (*PEM*, *PEM-MIME* и т.д.).

RSA многие годы противостоит интенсивному криптоанализу. Криптостойкость *RSA* основана на трудоемкости разложении на множители (факторизации) больших чисел. Открытый и секретный ключи являются функциями двух больших (100 ~ 200 двоичных разрядов или даже больше) простых чисел. Предполагается, что задача восстановления открытого текста по шифротексту и открытому ключу эквивалентна задаче факторизации.

Для генерации парных ключей используется два больших случайных простых числа, p и q. В целях максимальной криптостойкости p и q выбираются равной длины. Затем вычисляется произведение: $n = pq$.

Далее случайным образом выбирается ключ шифрования e, такой, что e и $\varphi(n) = (p-1)(q-1)$ являются взаимно простыми числами. Наконец расширенный алгоритм Евклида используется для вычисления ключа дешифрования d, такого, что $ed = 1 \bmod \varphi(n)$. Другими словами, $d = e^{-1} \bmod \varphi(n)$.

Заметим, что d и n – так же взаимно простые числа. Числа e и n – открытый ключ, а d – секретный. Два простых числа p и q хранятся в секрете. Для шифрования сообщения m необходимо выполнить его разбивку на блоки, каждый из которых меньше n (для двоичных данных выбирается самая большая степень числа 2, меньшая n). То есть если p и q – 100-разрядные простые числа, то n будет содержать около 200 разрядов. И каждый блок сообщения m_i должен иметь такое же число разрядов. (Если нужно зашифровать фиксированное число блоков, их можно дополнить несколькими нулями слева, чтобы гарантировать, что блоки всегда будут меньше n.) Зашифрованное сообщение c будет состоять из блоков c_i той же самой длины. Шифрование сводиться к вычислению $c_i = m_i^e \bmod n$. При дешифровании для каждого зашифрованного блока c_i вычисляется $m_i = c_i^d \bmod n$. Последнее справедливо, так как $C_i^d = (m_i^e)^e = m_i^{ed} = m_i^{k\varphi(n)+1} = m_i * m_i^{k\varphi(n)} = m_i * 1 = m_i$.

Все вычисления выполняются по модулю n.

Отметим, что сообщение может быть зашифровано с помощью d, а дешифровано с помощью e, возможен любой выбор.

Рассмотрим короткий пример. Если $p = 47$ и $q = 71$, то $n = pq = 3337$.

Ключ e не должен иметь общих множителей с $\varphi(n) = 46*70 = 3220$.

Выберем (случайное) e равным 79. В этом случае $d = 79^{-1} \bmod 3220 = 1019$. Опубликуем e и n, сохранив в секрете d. Для шифрования сообщения $m = 6882326879666683$ сначала разобьем его на блоки. Для выбранных параметров ограничимся блоками по три десятичных разряда. Сообщение разбивается на шесть блоков m_i: $m_1 = 688$; $m_2 = 232$; $m_3 = 687$; $m_4 = 966$; $m_5 = 668$; $m_6 = 003$

Первый блок шифруется как $688^{79} \bmod 3337 = 1570 = c_1$. Выполняя те же операции для последующих блоков, создадим шифротекст сообщения: $c = 1570275620912276242 3158$.

Для дешифрования нужно выполнить возведение в степень, используя ключ дешифрования 1019: $1570^{1019} \bmod 3337 = 688 = m_1$. Аналогично восстанавливается оставшаяся часть сообщения.

Эффективность реализации

Существует много публикаций, затрагивающих тему аппаратных реализаций *RSA*. Быстродействие аппаратной реализации *RSA* примерно в 1000 раз ниже, чем быстродействие аппаратной реализации *DES*. Быстродействие СБИС-реализации *RSA* с 512-битовым модулем – 64 Кбит/сек. Существуют также микросхемы *RSA*, которые оперируют с 1024-битовыми числами. В настоящее время разрабатываются микросхемы, которые, используя 512-битовый модуль, приблизятся к рубежу 1 Мбит/сек. Производители так же реализуют *RSA* в интеллектуальных карточках, однако производительность этих реализаций невысока. Программная реализация *DES* примерно в 100 раз быстрее программной реализации *RSA*. Эти оценки могут незначительно могут незначительно меняться в зависимости от используемых технологий, но *RSA* никогда не достигнет производительности симметрических алгоритмов.

Шифрование *RSA* выполняются намного эффективнее, если правильно выбрать значение e. Чаще всего используются 3, 17 или $65537 = 2^{16}+1$ – двоичное представление этого числа содержит только две единицы, поэтому для возведения в степень нужно выполнить лишь 17 умножений. Стандарт *X*.509 рекомендует число 65537, *PEM* – 3, *PKCS*#1 – 3 или 65537. Использование в качестве e любого из указанных значений (при условии, что сообщение дополняются случайными числами) не влияет на криптостойкость, даже если одно и то же значение *e* используется группой пользователей. Операции с секретным ключом можно ускорить при помощи китайской теоремы об остатках, если сохранить значения *p* и *q*, а так же заранее по секретному и открытому ключам вычислить вспомогательные значения: $d \bmod (p-1)$, $d \bmod (q-1)$ и $q^{-1} \bmod p$.

Криптостойкость *RSA*

Предполагается, что криптостойкость *RSA* связана с решением задачи разложения на множители больших чисел. Однако не было строго доказано, что нужно разложить *n* на множители, чтобы восстановить *m* по *c* и *e*. Не исключено, что может быть открыт совсем иной способ криптоанализа *RSA*. Однако если этот новый способ позволит криптоаналитику получить *d*, он также может быть использован для разложения на множители больших чисел. Так же можно атаковать *RSA*, определив значение $(p-1)(q-1)$. Однако этот метод не проще разложения n на множители. Доказано, что при использовании *RSA* раскрытие даже нескольких битов информации по шифротексту не легче, чем дешифрования всего сообщения. Самой очевидной атакой на *RSA* является разложение *n* на множители. Любой противник сможет получить

открытый ключ e и модуль n. Чтобы найти ключ дешифрования d, противник должен разложить n на множители. Криптоаналитик может перебирать все возможные d, пока не подберет правильное значение. Но подобная силовая атака даже менее эффективна, чем попытка разложения n на множители. В 1993 г. Был предложен метод кирптоанализа, основанный на малой теореме Ферма. К сожалению, этот метод оказался медленнее разложения на множители. Существует еще одна проблема. Большинство общепринятых тестов устанавливает простоту числа с некоторой вероятностью. Что произойдет, если p или q окажется составным? Тогда у модуля n будет три или более делителей. Соответственно некоторые делители будут меньше рекомендованной величины, что, в свою очередь, открывает возможности для атаки путем факторизации модуля. Другая опасность заключается в генерации псевдопростых чисел (чисел Кармайкла), удовлетворяющих тестам на простоту, но при этом не являющихся простыми. Однако вероятность генерации таких чисел пренебрежимо мала. На практике, последовательно применяя набор различных тестов на простоту, можно свести вероятность генерации составного числа до необходимого минимума.

Итоги по безопасности. На основании известных атак можно сформулировать следующие ограничения при использовании *RSA*:

- знание одной пары показателей шифрования/дешифрования для данного модуля позволяет злоумышленнику разложить модуль на множители;
- знание одной пары показателей шифрования/дешифрования для данного модуля позволяет злоумышленнику вычислить другие пары показателей, не расладывая модуль на множители;
- в криптографических протоколах с использованием *RSA* общий модуль использоваться не должен. (Это является очевидным следствием предыдущих двух пунктов.);
- для предотвращения раскрытия малого показателя шифрования сообщения должны быть дополнены («набиты») случайными значениями;
- показатель дешифрования должен быть большим.

Отметим, что недостаточно использовать криптостойкий алгоритм; безопасной должна быть вся криптосистема, включая криптографический протокол. Слабое место любого из трех компонентов сделает небезопасной всю систему.

Криптосистема Эль Гамаля

Криптосистему, предложенную Эль Гамалем в 1985 г. [9]можно использовать как для цифровых подписей, так и для шифрования. Криптостойкость определяется трудоемкость вычисления дискретного алгоритма в конечном поле. Криптоалгоритм не запатентован, но попадает под действие патента на метод экспоненциального ключевого обмена Диффи-Хеллмана. Данная система является альтернативой *RSA* и при равном значении ключа обеспечивает ту же криптостойкость. В отличие от *RSA* метод Эль Гамаля основан на проблеме дискретного логарифма. Если возводить число в степень в конечном

поле достаточно легко, то восстановить аргумент по значению (то есть найти логарифм) довольно трудно.

Для генерации пары ключей сначала выбираются простое число p и два случайных числа g и x; оба этих числа должны быть меньше p. Затем вычисляется $y = g^x \bmod p$.

Открытым ключом являются y, g и p. И g, и p можно сделать общими для группы пользователей. Секретным является x.

Вычисление и проверка подписи

Чтобы подписать сообщение M, сначала выбирается случайное число k, взаимно простое с p-1. Затем вычисляется $a = g^k \bmod p$, и с помощью расширенного алгоритма Евклида из уравнения $M = (xa+kb) \bmod (p\text{-}1)$ находиться b. Подписью является пара чисел: a и b. Случайное значение k должно храниться в секрете. Для проверки подписи необходимо убедиться, что $y^a a^b \bmod p = g^M \bmod p$.

Каждая новая подпись требует нового значения k, и это значение должно выбираться случайным образом. Если злоумышленник раскроет k, используемое Алисой, он сможет раскрыть секретный ключ Алисы x. Если злоумышленник сможет получить два сообщения, подписанные при помощи одного и того же k, он сможет раскрыть x, даже не зная k.

Рассмотрим простой пример. Выберем $p = 11$ и $g = 2$. Пусть секретный ключ $x = 8$. Вычислим

$$y = g^x \bmod p = 28 \bmod 11 = 3.$$

Открытым ключом являются $y = 3$, $g = 2$ и $p = 11$. Чтобы подписать $M = 5$, сначала выберем случайное число $k = 9$. Убедимся, что gcd(9,10) = 1. Далее вычислим

$$a = g^k \bmod p = 29 \bmod 11 = 6,$$

и затем с помощью расширенного алгоритма Евклида найдем b из уравнения

$$5 = (8*6+9*b) \bmod 10.$$

Решение: $b = 3$, а подпись представляет собой пару: $a = 6$ и $b = 3$. Для проверки подписи убедимся, что $y^a a^b \bmod p = g^M \bmod p$:

$$3663 \bmod 11 = 25 \bmod 11.$$

Шифрование/дешифрование

Некоторая модификация позволяет использовать криптосистему для шифрования/дешифрования сообщений.

Для шифрования сообщения M сначала выбирается случайное число k, взаимно-простое с p-1. Затем вычисляются:

$$a = g^k \bmod p,\ b = y^k M \bmod p.$$

Пара (a, b) является шифротекстом. Отметим, что шифротекст в два раза длиннее открытого текста.

$$M = \frac{b}{a^x} \bmod p.$$

Для дешифрования (a, b) вычисляются. Описанное преобразовании это то же самое, что и экспоненциальный ключевой обмен по Диффи-Хеллману,

за исключением того, что обмен по Диффи-Хеллману, за исключением того, что это часть ключа, а при шифровании сообщение умножается на y^k.

Криптосистема Диффи–Хеллмана

Первый из опубликованных алгоритмов на основе открытых ключей появился в работе Диффи и Хеллмана [8]. Обычно этот алгоритм называют обменом ключами по схеме Диффи–Хеллмана. Данная технология

обмена ключами реализована в целом ряде коммерческих продуктов. Цель данной схемы – предоставить двум пользователям защищенную возможность обменяться ключами. Сам по себе алгоритм ограничивается процедурой обмена ключами. В этой схеме имеются два открытых для всех числа: простое число q и целое число a – первообразный корень q3. Для обмена ключами пользователи А и Б производят следующие вычисления.

1. Пользователи A и B выбирают случайные целые числа XA и $XB < q$ – секретные ключи пользователей.

2. Пользователи A и B вычисляют открытые ключи пользователей $YA = aX, A \bmod q$ и $YB = aX, B \bmod q$, открытые ключи публикуются.

3. Вычисляется общий секретный ключ:

для пользователя A: $K = (YB)XA \bmod q$,

для пользователя B: $K = (YA)XB \bmod q$,

очевидно, что ключи, вычисленные пользователями, одинаковы.

Криптосистема на основе эллиптических уравнений

Эллиптические кривые - математический объект [5], который может определен над любым полем (конечным, действительным, рациональным или комплексным). В криптографии обычно используются конечные поля. Эллиптическая кривая есть множество точек (x, y), удовлетворяющее следующему уравнению: $y2 = x3 + ax + b$, а также бесконечно удаленная точка. Для точек на кривой довольно легко вводится операция сложения, которая играет ту же роль, что и операция умножения в криптосистемах *RSA* и Эль-Гамаля.

В реальных криптосистемах на базе эллиптических уравнений используется уравнение: $y2 = x3 + ax + b \bmod p$, где $p > 2^{255}$ – простое число; $a, b \in Fp$ – коэффициенты эллиптической кривой; Fp – конечное простое поле, представляемое как множество из p целых чисел $\{0, 1, ..., p - 1\}$; $4a^3 + 27b^2$ несравнимо с нулем по модулю p.

Проблема дискретного логарифма на эллиптической кривой состоит в следующем: дана точка G на эллиптической кривой порядка r (количество точек на кривой) и другая точка Y на этой же кривой. Нужно найти единственную точку x такую, что $Y = xG$, то есть Y есть x-я степень G.

Для шифрования выбираются хешфункция $h(.)$, уравнение эллиптической кривой E и образующая точка G кривой E.

Секретным ключом является показатель l.

Открытым ключом является точка кривой $Y = l*G$.

Алгоритм шифрования имеет следующий вид:

1. Для зашифровывания сообщения m, $0 \leq m \leq q - 1$,

где q – простое число, порядок циклической подгруппы группы точек эллиптической кривой E, отправитель генерирует целое число k, $2 \leq k \leq q - 1$.

2. Вычисляется точка $R = k*G$, значение хеш-функции $h(k*Y)$ и шифрограмма $c \equiv m + h(k*Y) \bmod q$.

3. Зашифрованный текст представляет собой пару (R, c).

Для расшифровывания получатель вычисляет точку $l*R = l*k*G = k*Y$, находит значение $h(k*Y)$ и находит сообщение $m \equiv c - h(l*R) \bmod q$.

Исследования в области эллиптических кривых показали, что криптосистемы, построенные на их основе, являются конкурентами по отношению к другим асимметричным криптосистемам, так как при эквивалентной стойкости используют ключи меньшей длины и имеют большую производительность. Более того, современные реализации показывают, что эти системы гораздо более эффективны, чем другие системы с открытыми ключами. Их производительность приблизительно на порядок выше, чем производительность *RSA*, Диффи–Хеллмана и *DSA*.

Области применения криптосистем с открытым ключом

Таблица

Система	Шифрование	Цифровая подпись	Обмен ключами
RSA	Да	Да	Да
Диффи- Хеллмана	Нет	Нет	Да
Эль Гамаля	Да	Да	Да
Эллиптические кривые	Да	Да	Да

Выводы. Использование асимметричних криптографических систем не требует передачи секретного ключа, который может быть перехвачен нарушителем. Из-за низкой скорости такие криптосистем их использую в основном для цифровой подписи и шифрования секретных ключей

Список литературы: 1. ГОСТ Р 34.10–2001. Информационная технология.Криптографическая защита информации. Процессы формирования и проверки электронной цифровой подписи. –М.: Госстандарт России. **2.** *Ростовцев А. Г., Михайлова Н. В.* Методы криптоанализа классических шифров. –М.: Наука, 1995. –208 с. **3.** *Диффи У.* Первые десять лет криптографии с открытым ключом //ТИИЭР, т. 76(1988)б Т5б с. 54-74. **4.** *Жельников В.* Криптография от папируса до компьютера. – М.: ABF, 1997. – 336 с. **5.** *Романец Ю. В., Тимофеев П. А., Шаньгин В. Ф.* Защита информации в компьютерных системах и сетях/ Под ред. В. Ф. Шаньгина. – М.: Радио и связь, 1999. – 328 с. **6.** *Саломаа А.* Криптография с открытым ключом. –М.: Мир, 1996. – 318 с. **7.** *Столингс В.* Основы защиты сетей. Приложения и стандарты / Пер. с англ. – М.: Издательский дом «Вильямс», 2002. – 432 с. : ил. **8.** ***Diffie W. and Hellman M.*** New directions in cryptography // IEEE Transactions on Information Theory. – November, 1976. **9.** ***ElGamal T.*** A public key cryptosystem and a signature scheme based on discrete logarithms //IEEE Transactions on Information Theory. – 1985. – V. IT-31. – π 4. – P. 469–472. **10.** *Баричев С. В.* Криптография без секретов. –М.: Наука, 1998. –120 с.

Поступила в редколлегию 08.11.2008